LETTER

A compact quick-start sub-mW pulse-width-controlled PLL with automated layout synthesis using a place-and-route tool

Wang Jing^{1a)}, Tetsuya Iizuka^{1,2}, Zule Xu², and Toru Nakura³

Abstract This paper demonstrates the design flow of a quick-start pulse-width-controlled PLL with automated layout synthesis using a place-and-route tool. The quick-start PWPLL converts the internal state into an analog-digital mixed signal called soft-thermometer-code (STC) and stores them into memory before PWPLL is turned off in order to enhance the start-up in the next turn-on. Our chip fabricated with TSMC 65 nm shows 220 ns settling time (13 reference clock cycles), 858 μ W power consumption under 1 V nominal supply voltage with 59 μ m $\times$ 58 μ m silicon area. The measurement results demonstrate that the design-automated PLL realizes the FoM of -221.7 dB, which is roughly the same value as that of the manually-designed one with the same target specification.

Keywords: phase locked loop, pulse-width-controlled PLL, quick start, layout automation, place and route

Classification: Integrated circuits

1. Introduction

Phase Locked Loop (PLL) has been one of the critical building blocks in modern communication systems, which synchronizes an output signal with a reference signal in frequency as well as in phase. PLL is also widely used in the field of power systems and power electronics, e.g. custom power controllers, flexible AC transmission systems, motor control, grid synchronization and uninterruptible power systems [1, 2, 3, 4, 5, 6, 7]. PLL can provide instantaneous detection and accurately track the voltage phase, which is vitally important in power systems. It is also used in a variety of signal estimation applications such as peak, frequency, phasor, harmonics, intermodulations, active and reactive power, sequence components, interference and transients [8, 9, 10, 11, 12].

A pulse width controlled PLL (PWPLL) works in the time domain and it is composed of mostly digital circuits [11]. The PWPLL is superior to other PLL structures in many aspects, such as its area efficiency and friendliness to the lowered power supply voltage, and is especially suitable for standard-cell-based design that enables automatic layout generation using a place-and-route (P&R) tool [13]. This is actually a good property of the PWPLL because

Fig. 1. Block diagram of the quick start PWPLL.

analog circuit building blocks, which are typically designed manually, often request much more design burden. For a typical analog component, its layout has to be manually re-designed every time the specification changes even slightly. Recently, synthesizable or automated designs of analog building blocks have been widely studied in order to relax the burden of manual analog designs [14, 15, 16, 17, 18, 19, 20, 21, 22, 23]. In addition, in some applications such as IoT (Internet of Things) [24], these devices are typically highly duty-cycled, hence sleep and startup power consumption often dominates the total power budget. Thus quick-start and low power consumption [25] of the PLL are crucial issues.

In this research, therefore, we aim to achieve automated layout synthesis of quick-start PWPLL and to implement quick start PWPLL with P&R tool in order to relax its design burden.

2. Quick-start PWPLL

As shown in Fig. 1, the PWPLL mainly consists of three parts, a PFD (Phase Frequency Detector), a PWCO (Pulse Width Controlled Oscillator) and a divider [11]. At every reference cycle, phase difference is translated by the PFD into a pulse, and the pulse width is converted by the TSTC (Time to Soft Thermometer code Converter) to tune the frequency of the R.OSC (Ring Oscillator). The STC (Soft Thermometer Code) is an N -bit thermometer code but its 1-bit is an analog value in between logic 0 and 1. The position of this analog bit depends on the locked frequency and phase.

A quick start method was originally proposed in [26]. The principle of the quick start function is briefly explained as follows: the internal state of the locked condition is digitized and stored before the PWPLL goes into its sleep mode, then restores the stored data so that the PWPLL can start from the state close to lock [26]. The PWPLL used D

¹Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo 113–0032, Japan

²VLSI Design and Education Center, The University of Tokyo, Tokyo 113–0032, Japan

³Department of Electronics Engineering and Computer Science, Fukuoka University, Fukuoka 814–0180, Japan

a) jwang@silicon.u-tokyo.ac.jp

DOI: 10.1587/eleex.16.20190546

Received August 26, 2019

Accepted September 3, 2019

Publicized September 17, 2019

Copied October 10, 2019

Fig. 2. (a) Schematic and (b) timing diagram of TSTC.

flip-flops as memory circuits, and used a TDC (time-to-digital converter) and a DTC (digital-to-time converter) to digitize and memorize the phase of the locked condition.

Since most building blocks of the PWPLL are designed with standard cells, it is highly motivated to implement the design in an automated P&R flow to reduce the design time and thus the cost. However, to realize such implementation, the generally-custom-designed passgates and capacitors in the PWCO including the TSTC and the dedicated ring oscillator need to be incorporated to the automated P&R flow. In this work, we propose the implementation method for these cells, and improve these building blocks, enabling a fully automated P&R, fast-lock, and small-area PLL.

3. Circuit design

3.1 Requirements for the PWCO

A schematic and a timing diagram of the TSTC in the PWCO are shown in Fig. 2. The FED (Fall Edge Detector) in Fig. 2(a) generates a short pulse at the end of the input pulse, which is used to sample the voltage V_{ak} . Then V_{ak} are held as the STC as shown in Fig. 2(b). We make that 1-bit analog value in STC by sampling and holding the output voltage of the slow inverter on the falling edge. In order for the PWPLL to be implemented with an automatic P&R tool, we need to prepare dedicated slow inverter and passgate cells, as highlighted in red in Fig. 2(a).

The analog value in the STC changes between VDD and GND, resulting in a mild change in the output frequency. We designed the analog value in the STC to be only one or two bit. To ensure there is at most two analog values in STC, the falling edge of the slow inverter of two adjacent stages of the TSTC must have an overlap. We use $t_{invslow}$ and τ_{Delay} to indicate the fall time of the slow inverter and the delay time of the buffers of each TSTC stage, respectively. As shown in Fig. 3(a), there will be no analog value in STC if $t_{invslow} < \tau_{Delay}$. Therefore, we set $t_{invslow} > \tau_{Delay}$ so that there is at least one analog value in STC. In addition, to ensure that there is at most two analog values in STC, $t_{invslow} < 2\tau_{Delay}$, as shown in Fig. 3(b).

According to [27], in addition, a locked PWPLL has to satisfy

Fig. 3. Timing diagram of TSTC when the relationship between τ_{Delay} and $t_{invslow}$ changes: (a) $t_{invslow} < \tau_{Delay}$ and (b) $t_{invslow} > 2\tau_{Delay}$.

Fig. 4. Input pulse width vs STC

$$0 < K_{PWCO} < 4\pi f_{out}, \quad \text{where} \quad K_{PWCO} \triangleq \frac{2\pi}{f_{ref}} \cdot \frac{\partial f_{out}}{\partial PW}. \quad (1)$$

f_{ref} and f_{out} are reference and output frequencies of the PWPLL, and PW is the input pulse width to the PWCO. $\partial f_{out} / \partial PW$ expresses the rate of change of f_{out} with PW . Under an ideal situation where f_{out} is a linear function of PW , $\partial f_{out} / \partial PW$ has a constant value, which is equal to $\Delta f_{out} / \Delta PW$ where Δf_{out} and ΔPW are the ranges of f_{out} and PW , respectively. The relationship between PW and STC is shown in Fig. 4. ΔPW is defined by the time difference between two PW s that makes STC all 1 and that makes it all 0, which is equal to $(N-1) \cdot \tau_{Delay} + t_{invslow}$. In summary, to ensure that the PWPLL can lock, the following two relationships must be satisfied:

$$\tau_{Delay} < t_{invslow} < 2\tau_{Delay} \quad (2)$$

$$\frac{1}{2} \frac{\Delta f_{out}}{f_{ref} \cdot f_{out}} < (N-1)\tau_{Delay} + t_{invslow} \quad (3)$$

In order to satisfy the condition (2) and (3), we designed τ_{Delay} to be 5 ns by adjusting the number of buffer cells, and designed $t_{invslow}$ to be 6 ns, which will be realized with the custom-designed inverter cell.

Though the above discussion assumes the ideal situation where K_{PWCO} has a constant value, in practice the relationship between f_{out} and PW is not linear and has a curve like as shown by the green line in Fig. 5(b). Thus as shown in Fig. 5(c), within the output frequency range

Fig. 5. (a) The slope of V_{ak} , (b) Relationship between PW and f_{out} and (c) Relationship between PW and K_{PWCO} .

K_{PWCO} may have larger value than $4\pi f_{out}$, which violates (1). To satisfy (1) all over the target output frequency range, the number of series passgates in the TSTC as in Fig. 2(a) has to be carefully chosen as it has an impact on PWCO linearity as summarized in Fig. 5. Fig. 5(a) and (b) respectively show the conceptual waveforms of V_{ak} and the relationship between the output frequency and the input pulse width when we change the number of passgates.

As shown by the purple lines in Fig. 5(a) and (b), if the transition of V_{ak} is too fast, the output frequency will have almost discrete trend to PW because the STC code rarely has an analog value between 0s and 1s. When we make the transition of V_{ak} slower by increasing the number of passgates as shown by the green and black lines in Fig. 5(a), the output frequency changes more smoothly as the STC becomes more likely to have an analog value. Then the relationship between the input pulse width and the output frequency will be closer to the ideal situation as shown in Fig. 5(b). The dotted line in Fig. 5(c) represents the ideal situation where the PWCO has constant K_{PWCO} . The red line indicates the value of $4\pi f_{out}$ as in (1). The PWPLL will lose lock at the frequencies where K_{PWCO} is above the red line because the condition (3) is not satisfied. When we use slow transition in V_{ak} the PWPLL can lock all over the target range as shown by the black line in Fig. 5(c). By satisfying this criteria, we can choose the appropriate number of series passgates to design the TSTC.

3.2 Frequency range of the ring oscillator

The schematic of our ring oscillator is shown in Fig. 6. Its frequency is tuned by the STC by changing the load seen from each inverter stage using the passgate and the capacitor cells as in the TSTC. We use these cells to tune the output frequency range of the oscillator. The effect of the capacitance C_L and parallel passgates on the relationship between the STC and output frequency is illustrated in Fig. 7. C_L actually changes only the lower limit as shown in Fig. 7(a), because it determines the maximum load capacitance seen from the inverters in the ring oscillator. On the other hand, the number of parallel passgates changes both the upper and lower limits as shown in

Fig. 6. Schematic diagram of the ring oscillator in the PWCO.

Fig. 7. Relationship between the STC and output frequency depending on (a) the capacitance C_L and (b) the number of parallel passgates.

Fig. 7(b), because the ON resistance and parasitic capacitance of the passgate change the impact of C_L on the inverter delay. Thus we can first fix the number of parallel passgates to meet upper limit of the output range, then tune C_L to meet the lower limit.

3.3 Custom cell designs

Based on the discussion in the previous subsections, we need to introduce the passgate and capacitor cells to the automated P&R flow. As the analog cells such as passgates and capacitors are typically not included in a standard-cell library, we have to add custom cells to the library. The schematic and layout images of our additional passgate and capacitor cells are shown in Fig. 8(a) and (b), respectively. Both the cells are designed by modifying the smallest inverter in the library so that we can easily realize these custom cells with P&R compatibility. The passgate cell is realized by changing the connection between the PMOS and NMOS devices so that they are connected in parallel. The capacitor cell is realized only with the NMOS gate capacitance by removing the PMOS device in order to have fine capacitance tunability. When the specification changes, we can adjust the number of these cells to meet the design conditions instead of re-designing these cells.

In addition, to realize the 6ns transition time of the slow inverter that is required for the TSTC, we investigated two possible ways: one is to add capacitors at the output of a standard inverter cell and the other is using a custom inverter cell by changing the gate length L_s . Because the former way will significantly increase the area than the latter, we chose to make a custom inverter cell with large L_s as shown in Fig. 8(c). This custom cell is also designed based on the smallest inverter in the library by increasing its L by 20 times. As this modification changes the cell width, we need to take care of the minimum tile width for the P&R compatibility for this customization. Fortunately, even when the design specification changes, we can fix $t_{invslow}$ and adjust τ_{Delay} , which is realized with standard

Fig. 8. Schematic and layout images of additional cells with the pitch and power straps as same as those of the standard cells. (a) passgate cell, (b) capacitor cell and (c) slow inverter cell.

buffer cells, to meet the condition (2) instead of re-designing a new custom inverter cell.

4. Measurement results

Our quick-start PWPLL was designed and fabricated using TSMC 65 nm standard CMOS process. We used Synopsys *IC-Compiler* [28] as a P&R tool to automatically generate the layout of the PWPLL. The chip micrograph is shown in Fig. 9, where the core occupies $59\mu\text{m} \times 58\mu\text{m}$. The target specification for the fabricated PWPLL is set to be the same as that of the manually-designed one [26]. Though the manually-designed chip in [26] was fabricated using 180 nm process, we re-designed and fabricated the manually-designed chip with TSMC 65 nm process for a fair performance comparison.

Fig. 9. Chip micrograph. The core area is $59\mu\text{m} \times 58\mu\text{m}$.

Fig. 10. Block diagram of the measurement setup.

We measured two PWPLLs in the same measurement environment. The measurement setup is illustrated in Fig. 10. The control signals are generated by Arduino uno. The reference clock is fed by Tektronix AFG 31000 SERIES arbitrary function generator. We used Keysight B2901A to provide power and to measure the current. Keysight 86100D Infiniium DCA-X sampling oscillo-

Fig. 11. Locking process of our PWPLL. (a) Without preset and (b) With preset.

scope, N9030A PXA signal analyzer and DSO-X 93304Q real time oscilloscope are used to measure the jitter, phase noise and locking process of the PWPLL output signal, respectively.

The measured locking process is shown in Fig. 11. We defined the lock time of the PLL as the time until the median of the output frequency falls within 0.5% of the target value. According to this definition, the lock time is 570 ns without preset and 220 ns with preset. This demonstrates that the quick start functionality works appropriately. The jitter measurement result is shown in Fig. 12, which shows $\sigma = 8.89\text{ ps}$. Measured phase noise of $L_{PWPLL} = -103.3\text{ dBc/Hz}@1\text{ MHz}$ is shown in Fig. 13.

Fig. 12. Jitter measurement result. $\sigma = 8.89\text{ ps}$.

Fig. 13. Phase noise measurement result. $L_{PWPLL} = -103.3\text{ dBc/Hz}@1\text{ MHz}$.

The performance comparison between two PWPLLs is summarized in Table I. The nominal output frequency of both the PWPLLs is designed to be 960 MHz. The lock time of the P&R PWPLL is 220 ns (13 reference clock), which is 3 cycles slower than the manual PWPLL. The possible reason of this is that our PWPLL has slightly different loop gain from the manually-designed PWPLL

and affect the step response. The power of the P&R PWPLL is 858 μ W and the area is 3454 μ m², which are comparable to the manual PWPLL. As the core utilization was slightly relaxed in order to pass the design rule check (DRC), the area of our PWPLL is 12% larger than the manual PWPLL. A PLL FoM, which is defined as:

$$\text{FoM}_{PLL} = 10 \log \left[\left(\frac{\sigma_{t,PLL}}{1 \text{ s}} \right)^2 \cdot \frac{P_{PLL}}{1 \text{ mW}} \right], \quad (4)$$

is –221.7 dB for the P&R PLL. The similar FoM as that of the manually-designed PWPLL is realized as shown in Table I. The design man-hours of the two PWPLLs are also compared. The automated flow takes only a few hours to complete the design once the custom-designed cells are prepared, while the manual design takes roughly a week. Note that these numbers are based on this specific design experience and may vary depending on the cases. Especially in the case of the specification changes, the proposed flow relaxes the design burden a lot as it can be completed just by changing the design parameters given to the automated script. These results verify that even with the automated design flow, we can realize the quick-start PWPLL design with comparable area and power efficiency, while it drastically reduces the design burden.

The comparison of this work with other ring-oscillator-based PLLs is summarized in Table II. In this comparison, the proposed PWPLL achieved smallest area and faster

lock in. Note that although we used a relatively high reference frequency for faster lock in, we still achieved sub-mW power consumption. The jitter is acceptable to most applications that require less precise clock generation, and comparable with [31] and [32]. Thanks to the sub-mW power consumption, our PWPLL realizes competitive FoM compared to the prior works.

5. Conclusion

We designed the quick-start PWPLL with automated layout synthesis using a P&R tool to relax the burden of the layout design, and achieved almost the same performance as that of the manual design. Our PWPLL has advantages in area and power efficiency, and realizes competitive FoM compared with prior works.

Acknowledgments

This paper is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO) and is partly supported by JSPS KAKENHI Grant Number 17H03244.

The VLSI in this work was designed through the support of VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Synopsys, Inc., Cadence Design Systems, Inc., and Mentor Graphics, Inc.

References

- [1] M. Karimi-Ghartemani, *et al.*: “Derivation and design of in-loop filters in phase-locked loop systems,” *IEEE Trans. Instrum. Meas.* **61** (2012) 930 (DOI: 10.1109/TIM.2011.2172999).
- [2] M. Karimi-Ghartemani, *et al.*: “A signal processing system for extraction of harmonics and reactive current of single-phase systems,” *IEEE Trans. Power Deliv.* **19** (2004) 979 (DOI: 10.1109/TPWRD.2004.829942).
- [3] R. Santos Filho, *et al.*: “Comparison of three single-phase PLL algorithms for UPS applications,” *IEEE Trans. Ind. Electron.* **55** (2008) 2923 (DOI: 10.1109/TIE.2008.924205).
- [4] F. Xiong, *et al.*: “A novel PLL for grid synchronization of power electronic converters in unbalanced and variable-frequency environment,” *Proc. 2nd IEEE Int. Symp. PEDG* (2010) 466 (DOI: 10.1109/PEDG.2010.5545832).
- [5] A. Nagliero, *et al.*: “Synchronization techniques for grid connected wind turbines,” *Proc. 35th IEEE IECON* (2010) 4606 (DOI: 10.1109/IECON.2009.5414859).
- [6] S.-K. Chung: “A phase tracking system for three phase utility interface inverters,” *IEEE Trans. Power Electron.* **15** (2000) 431 (DOI: 10.1109/63.844502).
- [7] P. Rodriguez, *et al.*: “Decoupled double synchronous reference frame PLL for power converters control,” *IEEE Trans. Power Electron.* **22** (2007) 584 (DOI: 10.1109/TPEL.2006.890000).
- [8] M. Karimi-Ghartemani and M. Iravani: “Measurement of harmonics/inter-harmonics of time-varying frequencies,” *IEEE Trans. Power Deliv.* **20** (2005) 23 (DOI: 10.1109/TPWRD.2004.837674).
- [9] A. Sarkar and S. Sengupta: “Second-degree digital differentiator-based power system frequency estimation under non-sinusoidal conditions,” *IET Sci. Meas. Technol.* **4** (2010) 105 (DOI: 10.1049/iet-smt.2008.0138).
- [10] M. Karimi-Ghartemani and M. Iravani: “Robust and frequency-adaptive measurement of peak value,” *IEEE Trans. Power Deliv.* **19** (2004) 481 (DOI: 10.1109/TPWRD.2004.824764).
- [11] T. Nakura and K. Asada: “Low pass filter-less pulse width controlled PLL using time to soft thermometer code converter,” *IEICE*

Table I. Performance comparison of manual and P&R designs of PWPLL.

	Manual [26]	P&R
Architecture	PWPLL	PWPLL
Supply voltage	1.0 V	1.0 V
Ref. Freq. [MHz]	60	60
Output Freq. [MHz]	960	960
Lock Range [MHz]	755~1050	848~1321
Lock Time	170 ns (10CLKs)	220 ns (13CLKs)
Power [μ W]	882	858
Area [μ m ²]	3084	3454
Jitter [ps]	7.80	8.89
FoM [dB]	–222.7	–221.7
Design man-hours	~80	~2

Table II. Comparison of this work and other PLLs.

	[29]	[30]	[31]	[32]	This work
Architecture	Digital	Digital	Digital	Digital	PWPLL
Layout Flow	P&R	P&R	manual	P&R	P&R
Ref. Freq.	10 MHz	80 MHz	9 MHz	4 MHz	60 MHz
Output Freq.	2.5 GHz	2.06 GHz	2.45 GHz	600 MHz	960 MHz
Lock time	–	–	1 μ s	0.5 μ s	220 ns
Process [nm]	65	28	65	180	65
Area [μ m ²]	42000	4300	83000	75000	3454
Power [mW]	13.7	6.95	0.25	14.5	0.86
Jitter [ps]	3.2	2.13	58	9.37	8.89
FoM [dB]	–218.5	–225.0	–210.8	–209.0	–221.7

- Trans. Electron. **E95-C** (2012) 297 (DOI: [10.1587/transele.E95.C.297](https://doi.org/10.1587/transele.E95.C.297)).
- [12] M. Karimi-Ghartemani and M. Iravani: “Wide-range, fast and robust estimation of power system frequency,” *Electr. Power Syst. Res.* **65** (2003) 109 (DOI: [10.1016/S0378-7796\(02\)00223-7](https://doi.org/10.1016/S0378-7796(02)00223-7)).
 - [13] N. Tohge, *et al.*: “A pulse width controlled PLL and its automated design flow,” *IEEE ICECS* (2013) 5 (DOI: [10.1109/ICECS.2013.6815331](https://doi.org/10.1109/ICECS.2013.6815331)).
 - [14] D. Yang, *et al.*: “A fully synthesizable injection-locked PLL with feedback current output DAC in 28 nm FDSOI,” *IEICE Electron. Express* **12** (2015) 20150531 (DOI: [10.1587/elex.12.20150531](https://doi.org/10.1587/elex.12.20150531)).
 - [15] N. Ojima, *et al.*: “A synthesizable digital low-dropout regulator based on voltage-to-time conversion,” *IFIP/IEEE VLSI-SoC* **561** (2018) (DOI: [10.1109/VLSI-SoC.2018.8644879](https://doi.org/10.1109/VLSI-SoC.2018.8644879)).
 - [16] N. Ojima, *et al.*: “A 65 nm CMOS synthesizable digital low-dropout regulator based on voltage-to-time conversion with 99.6% current efficiency at 10-mA load,” *IFIP/IEEE VLSI-SoC* (2019) 1 (DOI: [10.1007/978-3-030-23425-6_1](https://doi.org/10.1007/978-3-030-23425-6_1)).
 - [17] H. C. Ngo, *et al.*: “8.5 A 0.42 ps-jitter –241.7 dB-FoM synthesizable injection-locked PLL with noise-isolation LDO,” *ISSCC* (2017) 150 (DOI: [10.1109/ISSCC.2017.7870305](https://doi.org/10.1109/ISSCC.2017.7870305)).
 - [18] W. Deng, *et al.*: “A 0.0066-mm² 780-uW fully synthesizable PLL with a current output DAC and an interpolative-phase coupled oscillator using edge injection technique,” *ISSCC* (2014) 266 (DOI: [10.1109/ISSCC.2014.6757428](https://doi.org/10.1109/ISSCC.2014.6757428)).
 - [19] W. Deng, *et al.*: “A 0.048-mm² 3-mW synthesizable fractional-N PLL with a soft injection-locking technique,” *ISSCC* (2015) 252 (DOI: [10.1109/ISSCC.2015.7063021](https://doi.org/10.1109/ISSCC.2015.7063021)).
 - [20] W. Kim, *et al.*: “A 0.032 mm² 3.1 mW synthesized pixel clock generator with 30 psrms integrated jitter and 10-to-630 MHz DCO tuning range,” *ISSCC* (2013) 250 (DOI: [10.1109/ISSCC.2013.6487721](https://doi.org/10.1109/ISSCC.2013.6487721)).
 - [21] Y. Li, *et al.*: “A novel fully synthesizable all-digital RF transmitter for IoT applications,” *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **37** (2018) 146 (DOI: [10.1109/TCAD.2017.2684097](https://doi.org/10.1109/TCAD.2017.2684097)).
 - [22] D. Yang, *et al.*: “A 0.0055 mm² 480 μ W fully synthesizable PLL using stochastic TDC in 28 nm FDSOI,” *IEICE Trans. Electron.* **E99-C** (2016) 632 (DOI: [10.1587/transele.E99.C.632](https://doi.org/10.1587/transele.E99.C.632)).
 - [23] N. Ojima, *et al.*: “A 0.0053-mm² 6-bit fully-standard-cell-based synthesizable SAR ADC in 65 nm CMOS,” *IEEE NEWCAS* (2019).
 - [24] L. Atzori, *et al.*: “The Internet of Things: A survey,” *Comput. Netw.* **54** (2010) 2787 (DOI: [10.1016/j.comnet.2010.05.010](https://doi.org/10.1016/j.comnet.2010.05.010)).
 - [25] C.-C. Chung, *et al.*: “A 0.52/1 V fast lock-in ADPLL for supporting dynamic voltage and frequency scaling,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **24** (2016) 408 (DOI: [10.1109/TVLSI.2015.2407370](https://doi.org/10.1109/TVLSI.2015.2407370)).
 - [26] T. Nakura, *et al.*: “Quick-start pulse width controlled pll with frequency and phase presetting,” *IEICE Trans. Electron.* **E101-C** (2018) 218 (DOI: [10.1587/transele.E101.C.218](https://doi.org/10.1587/transele.E101.C.218)).
 - [27] T. Nakura, *et al.*: “A PLL compiler from specification to GDSII,” *IEICE Trans. Fundamentals* **E100-A** (2017) 2741 (DOI: [10.1587/transfun.E100.A.2741](https://doi.org/10.1587/transfun.E100.A.2741)).
 - [28] D. Lockhart, “Place and route using Synopsys IC compiler,” *ECE5745 Tutorial3*, January 30, 2016.
 - [29] Y. Park, *et al.*: “An all-digital PLL synthesized from a digital standard cell library in 65 nm CMOS,” *CICC* (2011) 1 (DOI: [10.1109/CICC.2011.6055347](https://doi.org/10.1109/CICC.2011.6055347)).
 - [30] M. Lee, *et al.*: “A 0.3-to-1.2 V frequency-scalable fractional-N ADPLL with a speculative dual-referenced interpolating TDC,” *ISSCC* (2018) 122 (DOI: [10.1109/ISSCC.2018.8310214](https://doi.org/10.1109/ISSCC.2018.8310214)).
 - [31] S. Hong, *et al.*: “A 250- μ W 2.4-GHz fast-lock fractional-N frequency generation for ultralow-power applications,” *IEEE Trans. Circuits Syst. II, Exp. Briefs* **64** (2017) 106 (DOI: [10.1109/TCSII.2016.2551598](https://doi.org/10.1109/TCSII.2016.2551598)).
 - [32] C.-T. Wu, *et al.*: “A two-cycle lock-in time ADPLL design based on a frequency estimation algorithm,” *IEEE Trans. Circuits Syst. II, Exp. Briefs* **57** (2010) 430 (DOI: [10.1109/TCSII.2010.2048358](https://doi.org/10.1109/TCSII.2010.2048358)).